

SAE - Algorithmme

Présentation :

Le programme simule la gestion d'une salle de sport, avec un fichier adhérent contenant toute les informations des différents adhérents tel que leur numéro de carte qui les identifie, leur nombre de points, la date de création de leur compte et l'état de leur carte néanmoins le fichier peut être vide si la salle de sport vient tout juste d'ouvrir. De plus, un fichier contenant les différentes activités est utilisé pour référencer leur numéro d'activité, leur coût en point pour qu'un adhérent puisse la pratiquer (forcément payante) et le nombre de personnes ayant participé à l'activité après le lancement du programme qui symbolise une nouvelle journée et encore le fichier peut être vide si la salle de sport est en cours de création.

Au lancement du programme l'utilisateur fait gagner des points au adhérent en fonction de la différence entre l'année actuelle (mise par défaut en 2022) et l'année de création de compte, le tout multiplié par 2. De plus, au chargement du fichier des activités symbolisant en quelque sorte une nouvelle journée, le nombre d'entrées est alors mise à 0.

L'utilisateur à au démarrage un menu qui s'affiche lui proposant dans la première page tout la gestion d'un compte, il peut obtenir les informations de tous les adhérents, celle d'un utilisateur en particulier, la création d'un compte ou bien résilié son compte si il est déjà client, mais aussi de choisir de pratiquer une activité.

Sans oublier la gestion de sa carte qui est un sous menu permettant à un adhérent en entrant son identifiant d'afficher ses propres informations, activer ou désactiver sa carte s' il l'a perdu, cassé, refaite, retrouvée par exemple. Bien sûr l'adhérent choisi peut alimenter sa carte en point pour effectuer des activités, pour cela 3 formules lui sont proposées à différents prix et différents avantages de points bonus. Et enfin l'utilisateur peut changer de carte pour alimenter le compte d'un autre adhérent par exemple.

Et pour finir, la 2e et dernière pages, consiste à la gestion des différentes activités. Tout d'abord l'affichage global de toutes les informations des activités ou alors l'affichage des informations d'une activité en particulier avec son numéro, et ensuite la création, la suppression, modification d'une activité.

Et enfin lorsque le l'utilisateur quitte le programme il a le choix de sauvegarder ou non les modifications qu'il vient d'apporter car on peut citer l'exemple d'une suppression d'activités ou d'un adhérent par erreur et de corriger ces erreurs.

Pour la structure, les adhérents et les activités ont un fonctionnement différent.

Tout d'abord les tableaux liés aux adhérents ont leur information triée dans l'ordre croissant des identifiant d'adhérent d'adhérents, sa taille physique est de 50 et sa taille logique dépend du nombre d'identifiant dans le fichier.

Alors que pour les activités les système est différent, car l'indice du tableau correspond au numéro d'activité, de plus les tableaux sont créés en étant remplis de 0 donc si le tableau tabCoutPoint à l'indice 34 par exemple est à 0 alors l'activité n'existe pas. Et donc la taille logique et physique sont les mêmes.

Adhérents	12	24	27	45	...
Points	2	1	18	7	...

Indices	0	1	2	3	4	5	6
Coût	10	0	0	22	17	0	4
Nombre Entrées	4	0	0	1	20	0	40

1. Description des variable/tableaux principale

tabAdhe[] : tableau de 50 entiers utilisé pour les identifiant des adhérent.

tabPoints[] : tableau de 50 entiers utilisé pour les différents nombres de points des adhérent.

jour[] : tableau de 50 entiers utilisé pour les jours de création des compte des adhérent.

mois[] : tableau de 50 entiers utilisé pour les mois de création des comptes des adhérent.

annee[] : tableau de 50 entiers utilisé pour la jour de création des compte des adhérent.

carte[] : tableau de 50 entiers utilisé pour les statuts des cartes des adhérents.

tailleLAdhe : variable d'entier contenant la taille logique du tableau tabAdhe[] soit le nombre d'éléments.

tabCoutPoint[] : tableau de 50 entier utilisé pour les prix des différentes activités (par rapport à l'indice qui correspond au numéro d'activité).

tabNbEntree[] : tableau de 50 entier utilisé pour les différents nombres d'entrée dans les différentes activités (par rapport à l'indice qui correspond au numéro d'activité).

tailleLAct : variable d'entier contenant la taille logique des tableaux en lien avec les activités, la taille logique et physique sont les même dans cette configuration

2. Description des fonctions

-Pour commencer nous avons 2 fonctions de chargement pour nos 2 fichiers "adherent.txt" et "activite.txt" qu'on utilise sous forme de tableau , celle-ci nous permettent à chaque lancement du programme d'intégrer ces fichiers afin que l'utilisateur puisse interagir avec , que ce soit les afficher ou les manipuler.

Les fichiers utilisés ont chacun pour fonction de stocker les données de différents utilisateurs , "adherent.txt" a pour but de stocker l'identifiant , le nombre de point auquel l'adhérent a accès , la date exacte à laquelle l'adhérent s'est inscrit et l'état actuelle de la carte qui est désactivé ou activé tandis que "activite.txt" contient le nombre de point nécessaire pour accéder à une activité , le nombre d'accès à cette même activité , et l'identifiant de l'activité .

-Pour nous permettre de rechercher ces mêmes données dans les différents fichiers , nous avons créé une fonction recherche nous permettant en fonction du tableau à rechercher de trouver la valeur que l'utilisateur recherche grâce à un pointeur "trouve" et la taille logique de ce dit tableau .

-La fonction décalage à droite permet la manipulation des tableaux , c'est une fonction qui sert à l'insertion de données par l'utilisateur , elle permet donc de décaler toutes les valeurs vers la droite à partir de la position d'insertion . Même idée pour la fonction décalage à gauche sauf que celle-ci sert à la suppression de données , celle-ci va décaler toutes les valeurs vers la gauche à partir de la position de la valeur que l'utilisateur souhaite supprimer.

-La fonction Ajouter est la concaténation de la fonction recherche et décalage à droite car celle-ci est la fonction d'insertion d'une valeur , pour ça elle se sert de la fonction recherche pour trouver la position à laquelle l'utilisateur veut insérer une nouvelle valeur , et puis de la fonction décalage à droite pour décaler toutes les valeurs suivantes et enfin permettre l'insertion dans les fonctions suivantes .

-Pour la suite direct nous avons la fonction Ajouter un adhérent qui fait appel à la fonction ajouter et qui permet d'ajouter toutes les informations d'un nouvel adhérent dans le fichier "adherent.txt" donc un nouveau numéro de carte ainsi que toutes les données qui le suivent.

-Tout comme la fonction ajouter , la fonction supprimer est une fonction qui recherche la position de la valeur que l'utilisateur souhaite supprimer grâce à la fonction de recherche et ensuite quand la position est trouvée alors toutes les valeurs subissent la fonction décalage à gauche ce qui permet donc la suppression d'une valeur dans un tableau comme dans la fonction suivante .

-supprimerAdherent utilise donc la fonction supprimer dans le tableau qui représente "adherent.txt" afin de permettre à l'utilisateur de rentrer l'identifiant de l'adhérent à supprimer.

-activerCarte est une fonction qui gère l'activation/désactivation d'une carte.
Elle prend en paramètre le tableau des adhérents, le tableau des cartes, la taille logique et la valeur choisie.
Elle active ou désactive la carte du compte choisie si elle est bien dans la base de données.

-gestionCarte est une fonction globalisée avec un menu permettant de gérer une carte individuellement.
Elle prend en paramètre toutes les variables lié au adhérent.
Fonction qui demande à l'utilisateur d'entrer un identifiant d'adhérent existant et lui affiche différentes options utilisables dans celle-ci.

-affichageTous est une fonction qui affiche la globalité des informations pour chaque adhérent.
Elle prend en paramètre toutes les variables lié au adhérent.
Fonction qui affiche les informations de tous les adhérents

-affichageCarte est une fonction d'affichage des informations d'un adhérent choisie.
Elle prend en paramètre tous les tableaux liés aux adhérents.

La fonction demande à l'utilisateur d'entrer un numéro d'adhérent en fonction de celui-ci affiche ses informations s'il est dans la base donnée.

-AffichageToutActivite est une fonction qui affiche les informations de toutes les activités présentes. Elle prend en paramètre un tableau contenant le coût de l'activité en fonction de son indice, un autre tableau contenant le nombre d'entrées d'une activité en fonction de son indice et sa taille logique. La fonction affiche les informations de toutes les activités en vérifiant que dans les différents tableaux, que l'activité coûte des points pour l'afficher c'est-à-dire qu'elle existe.

-AffichageActivite est une fonction permettant d'avoir les informations sur une activité en particulier. Celle-ci prend en paramètre un tableau contenant le coût de l'activité en fonction de son indice, un autre tableau contenant le nombre d'entrées d'une activité en fonction de son indice et sa taille logique. La fonction demande à l'utilisateur d'entrer un numéro d'activité si celui-ci n'a pas de point cela signifie que l'activité n'existe pas, l'utilisateur en est notifié et la fonction se termine mais si celle-ci contient des points alors une affichage avec ses différentes informations s'affiche tel que son numéro, son coût en point et son nombre d'entrées dans la journée.

-alimenterCarte est une fonction qui est utilisée pour acheter des points afin d'effectuer par la suite des activités, elle prend en paramètre un tableau avec les identifiants d'adhérents, un tableau contenant le nombre de points des adhérents, et une variable qui contient la taille logique soit le nombre d'éléments. l'utilisateur entre l'identifiant d'un adhérent et le programme vérifie qu'il est bien dans la base de données ensuite les différentes formules s'affichent proposant à l'utilisateur d'en choisir une, de plus l'entrée est vérifiée. Et enfin dès lors que la formule est choisie le compte est crédité de point en fonction de la formule choisie, et un récapitulatif de l'achat est affiché

-ajoutActivite permet d'ajouter une activité et son coût. Elle prend en paramètre le tableau contenant le coût de l'activité et la taille logique. Elle demande le numéro de la nouvelle activité puis vérifie si elle est déjà dans le tableau ou si celui-ci est plein si c'est le cas elle renvoie un message d'erreur et s'arrête. Sinon elle demande le coût de l'activité et renvoie un récapitulatif de l'activité pour finir elle actualise le coût de l'activité.

-suppActivite permet de supprimer une activité et son nombre de points. Elle prend en paramètre le tableau contenant le coût de l'activité, le tableau contenant le nombre d'entrées, et la taille logique. Elle demande le numéro de l'activité que l'on veut supprimer puis vérifie si elle existe et si le tableau est plein si c'est le cas elle renvoie un message d'erreur et s'arrête. Si l'activité existe elle demande si on veut vraiment le supprimer et tant que la réponse n'est pas oui("O") ou non ("N") la question est redemandée. Si le choix est non ("N"), "Annulation de la suppression" est affiché et la fonction s'arrête. Si le choix est oui ("O") "suppression Activé" est affiché et les tableaux contenant le coût de l'activité et du nombre d'entrées sont actualisés.

modifActivite est une fonction permettant de modifier le coût d'une activité choisie. Elle prend en paramètre un tableau contenant le coût de l'activité et une variable contenant la taille logique. La fonction demande à l'utilisateur d'entrer un numéro d'activité si celui-ci n'a pas de point cela signifie que l'activité n'existe pas, le programme affiche ensuite l'ancien coût de l'activité et propose à l'utilisateur de taper le nouveau prix qui doit être supérieur à 0 point, ensuite le prix est actualisé et un dernier affichage montre la dernière actualisation.

-AffichageActivite est une fonction permettant d'avoir les informations sur une activité en particulier. Celle-ci prend en paramètre un tableau contenant le coût de l'activité en fonction de son indice, un autre tableau contenant le nombre d'entrées d'une activité en fonction de son indice et sa taille logique. La fonction demande à l'utilisateur d'entrer un numéro d'activité si celui-ci n'a pas de point cela signifie que l'activité n'existe pas, l'utilisateur en est notifié et la fonction se termine mais si celle-ci contient des points alors une affichage avec ses différentes informations s'affiche tel que son numéro, son coût en point et son nombre d'entrées dans la journée.

-faireActivite est une fonction qui est utilisée pour simuler qu'un adhérent choisie, pratique une activité choisie avec la décrémentation de points en fonction du coût de l'activité et si l'adhérent a les ressources en point afin de pouvoir la payer. Elle prend en paramètre un tableau avec les identifiants d'adhérents, un tableau contenant le nombre de point des adhérent, un tableau contenant le coût de l'activité en fonction de son indice, un autre tableau contenant le nombre d'entrées d'une activité en fonction de son indice, la taille logique des adhérent et la taille logiques des Activité. La fonction demande d'entrer un numéro d'adhérent et vérifie si celui-ci est dans la base de données sinon un message d'erreur s'affiche et la fonction s'arrête ou encore si l'adhérent n'a pas fait d'activité dans la journée grâce au statut de sa carte. Ensuite l'utilisateur doit entrer un identifiant d'activité et tant qu'il n'est pas bon l'utilisateur doit en entrer un autre sauf s'il fait -1 alors la fonction s'arrête. Et pour finir la fonction vérifie que l'adhérent voulant pratiquer a assez de points alors son compte est débité et le nombre d'entrées augmente de 1 avec un affichage du nombre de points restant.

-SauvegardeAdherent et SauvegardeActivité sont des fonctions permettant de sauvegarder dans un fichier les informations dans différents tableaux liés aux adhérents. Elle prend en paramètre pour la SauvegardeAdherent un tableau contenant les identifiants des adhérents, un tableau contenant les points des différents adhérents, 3 tableaux : jour, mois, année, un tableau carte contenant le statut de la carte 1 pour activer et 0 pour désactiver et une variable contenant la taille logique. Et pour la SauvegardeActivite, elle prend en paramètre le tableau contenant le coût de l'activité, le tableau contenant le nombre d'entrées, et la taille logique.

Celle-ci ouvre le fichier correspondant à la fonction en écriture, renvoie -1 si il y a une erreur et 0 si tout est bon. Le fichier est scanné ligne par ligne jusqu'à la fin, insère les données dans les tableaux et scanne la ligne suivante et ainsi de suite jusqu'à avoir scanné tout le document et arrêter l'écriture.

-Sauvegarde est une fonction qui utilise les 2 fonctions précédentes pour savoir si l'utilisateur veut enregistrer les données dans les fichiers. Elle prend tous les tableaux liés aux adhérents, tous les tableaux liés aux activités et leur taille logique respective.

la fonction demande a l'utilisateur si il veut Sauvegardé ou non, si oui alors les 2 fonctions sont utilisé sinon une sécurité a été ajouter pour être sûr que la sauvegarde ne se fait pas ou c'est une erreur de frappe et que vous la sauvegarde est bel et bien faite

-Menu est une fonction d'affichage des différentes options du gestionnaire.
La fonction a une sécurité si l'entré est incorrecte et aussi elle possède 2 pages.

-GestionSalle est la fonction globale de notre programme, elle permet de gérer toutes les fonctions vu juste au dessus et d'utiliser toutes nos variables.
Elle fonctionne en exécutant les différentes options choisies par l'utilisateur durant son usage, jusqu'à ce qu'il choisisse l'option pour l'arrêter.

-clearpage est une fonction qui nettoie l'écran et demande à l'utilisateur de presser Entrée pour continuer.
Elle demande à l'utilisateur de presser une touche afin d'effacer le contenu de l'écran et de rendre le résultat plus propre et clair pour l'utilisateur.

3.Travaille effectué par le groupe

Fonctions :

Chargement Adhérent : Louis / Ana

Chargement Activité : Alexis

recherche : Louis

décalage Droite : Louis

décalage Gauche : Louis

ajouter : Louis

ajouter Adhérent : Louis

suppression : Louis

affichage Tous : Louis / Alexis

affichage Carte : Alexis / Louis

affichage Activités : Alexis

affichage Tout Activité : Alexis

alimenter Carte : Alexis / Sami

ajout Activité : Alexis

supp Activité : Alexis

faire Activité : Alexis

Sauvegarde Adhérent : Louis

Sauvegarde Activité : Alexis

supprimer Adhérent : Louis

modif Activité : Alexis

active Carte : Louis

gestionCarte : Louis

Sauvegarde : Louis

GestionSalle : Sami

Menu : Sami / Alexis

clearpage : Sami / Alexis

Cahier des charges

Ecriture cahier des Charge : Sami/Alexis/Ana
Doxygen / documentation : Louis

Coder et Tester

Coder : Louis / Alexis / Sami / Ana
Tester : Louis / Alexis / Sami / Ana